WEP
Warwick
Evans
Publishing

International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 2, Number 2, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v2n2.43
Journal homepage: https://wepub.org/index.php/IJCSIT/index

A Study on Tuberculosis CT Image Classification Based on Federated Learning Methods

Zhihan Yang[1, *], Xiaogang Wang[2]

[1]Chongqing University of Science and Technology, Chongqing, China;

[2]Radiology Department, The People's Hospital of Yuechi County, Sichuan, China.

[*]Corresponding Author: 932527760@qq.com

ABSTRACT

The widespread application of deep learning in the medical field has been increasingly prominent. The utilization of deep learning for the recognition and classification of tuberculosis (TB) in CT medical images has become a popular research topic. However, federated learning exhibits unique advantages. Besides accomplishing tasks that deep learning can perform, it also has the ability to protect patients' privacy to a certain extent. This characteristic makes it more readily acceptable. Nevertheless, in real-world scenarios, data heterogeneity among federated learning participants can lead to model bias, hindering the achievement of desired results. To address this challenge, this study proposes a method that employs client data distribution information clustering and a loss function suppression term. This approach demonstrates excellent performance in both data-heterogeneous and non-data-heterogeneous scenarios. Experimental validation on a TB CT image dataset confirms its effectiveness. Compared to traditional federated learning baseline methods such as Federated Averaging (FedAvg), our method achieves an improvement of up to 11.54% in model accuracy, with faster convergence speed. Moreover, it exhibits greater stability in model accuracy when subjected to communication cost constraints.

KEYWORDS

Federated Learning; Tuberculosis CT Images; Data Heterogeneity; Image Classification

1. INTRODUCTION

TB is one of the serious infectious diseases that poses a significant threat to human health and is considered a major global public health issue worldwide [1-2]. According to relevant statistics, the global number of TB infections has been approximately 20 million in recent years, with approximately 8 million new TB cases added each year [3]. Medical imaging information plays a critically important role in the diagnosis of TB [4]. The radiological diagnosis of pulmonary TB is typically carried out by experienced radiologists, and this process heavily relies on the expertise and skill level of the diagnosing physician.

Radiologists in the hospital carefully observe and analyze various radiographic features when analyzing pulmonary TB images, such as pulmonary consolidation, cavities, nodules, and others, These features exhibit a diversity of presentations in different cases, and the location, size, and morphology of pulmonary TB lesions can impact the physician's diagnostic judgment [5]. Furthermore, manual identification and analysis require a significant amount of time and cognitive effort, especially when dealing with a large volume of images that require urgent processing. Diagnostic fatigue and oversights can occur in such scenarios, making it challenging to ensure both efficiency and accuracy in clinical diagnosis. In recent years, addressing this practical issue, many researchers have been exploring the application of computer-aided diagnostic systems in clinical practice. These systems utilize deep learning and artificial intelligence technologies to automatically analyze and interpret CT images [6], assisting physicians in rapidly identifying lesion areas and providing diagnostic support. By mitigating human factors' interference and uncertainties, such research has the potential to reduce the risk of missed diagnoses and misdiagnoses in practical applications, thereby enhancing both the accuracy and efficiency of radiologists' diagnoses.

However, in practical scenarios, medical resources in some remote and rural areas are relatively scarce, and healthcare professionals often lack experience. This leads to a low detection rate of pulmonary TB and instances of missed diagnoses and misdiagnoses [7]. Moreover, acquiring a sufficient quantity and diversity of pulmonary TB CT imaging data poses numerous challenges, including issues related to privacy concerns and significant time costs. In contrast to CT imaging data from healthy individuals, there is typically a limited amount of imaging data available for TB patients. This imbalance in the quantity of imaging data between healthy individuals and TB patients creates a data imbalance issue. Additionally, CT imaging data falls under the category of personal privacy information and is subject to legal protections and restrictions. In the realm of deep learning, an uneven distribution of samples can introduce bias into model performance [8]. For instance, when the number of imaging samples from TB patients is significantly lower than that from normal individuals, the model may tend to bias its predictions toward the larger dataset of normal imaging data to improve overall accuracy. Furthermore, due to the minimal differences in features between CT imaging data of some TB patients and normal individuals, a limited amount of data may result in suboptimal model performance [9]. The model may only capture the features of a subset of TB patient imaging data, leading to poor performance when faced with TB imaging data outside the training set. Similarly, if the model learns features from a large quantity of normal lung images, it may exhibit a higher error rate when identifying images of TB patients. In summary, the lack of sufficient TB data samples or significant disparities in data distribution can prevent the model from accurately capturing and learning the features of TB imaging. This can result in less-than-ideal performance in real-world applications. Additionally, concerns about personal privacy information make it rare for different hospitals to be willing to share data, creating data barriers. Therefore, it is of paramount importance to explore how to construct a collaborative deep learning model using imaging data from multiple hospitals without compromising the privacy of TB patients' data.

Federated Learning (FL) [10] is an emerging distributed machine learning technique in recent years. It differs from traditional distributed machine learning by placing a stronger emphasis on preserving the privacy of participant user data and addressing issues related to global model aggregation. The core idea of FL is that participant clients only need to upload their local model parameters to a central server for global model aggregation, without exposing their local data. This approach effectively resolves the problem of data silos [11] and safeguards the privacy of participant data. As FL participants operate in diverse real-world environments, there are variations in the training data contributed by each client, which is known as data heterogeneity [12-13]. This heterogeneity manifests primarily in terms of data volume, data distribution, and the efficiency of communication between participants and the server. It can potentially lead to biases in the iterative process of global model training in FL, ultimately resulting in suboptimal performance across all participant local clients.

Therefore, to address the aforementioned issues, this study proposes a federated learning approach for lesion screening in tuberculosis CT images. With the aim of preserving the privacy of data from different regional hospitals and patients, deep learning models are trained to assist medical professionals in quickly and accurately screening lesion areas in the CT images of tuberculosis patients. Considering the heterogeneity of image data from different regions and hospitals, a weighted model aggregation method based on the data distribution characteristics of federated learning participants is designed. Additionally, a suppression term is added to the local model loss functions to reduce the degree of offset between local models and the global model. Experimental investigations and validations of the above methods have been conducted.

2. OVERVIEW OF RELEVANT KNOWLEDGE

2.1. The Fundamental Architecture of FL

The basic architecture of FL is illustrated in the following figure 1. When using FL, there is typically a central entity whose task is to aggregate information from multiple client-side local learner models. These clients can be mobile devices or IoT devices, among others.

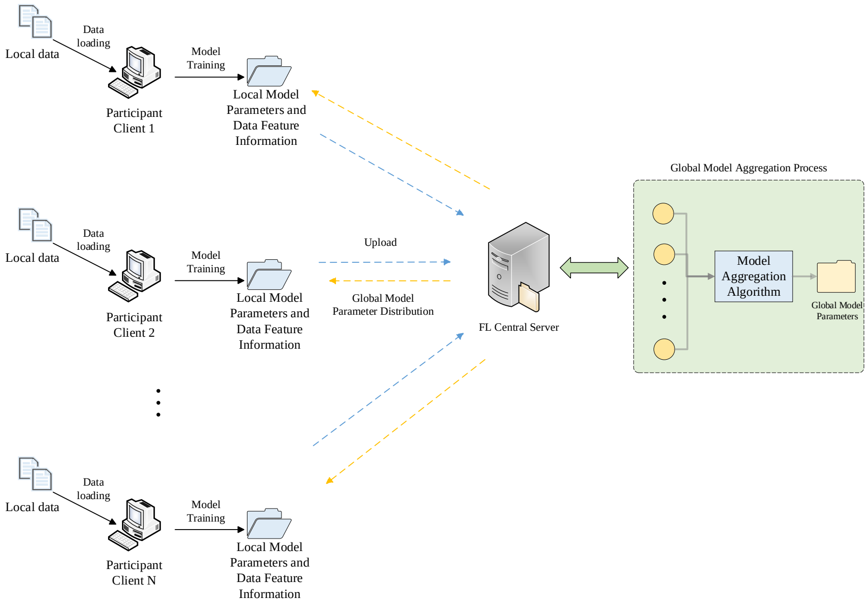

Figure 1. FL Basic Architecture Diagram

During the training process, the central entity initially sends the current state of the global model(referred to as w_t) to a small subset of clients. Then, each client utilizes its available data to perform the training process locally, aiming to minimize certain local objective functions related to $F_k(w_t)$.This process is designed to collaborate in order to enhance the global model while ensuring privacy and data security in local model training. After training, the models w_{t+1}^k from each client $k \in K$ are sent back to the central entity, and then the central entity aggregates all these local models in some way. This aggregation process employs the Federated Averaging (FedAvg) algorithm[14], which is one of the most classical federated learning model aggregation algorithms. This algorithm performs a data-weighted average of parameters from all models to generate a new global model w_{t+1}, as shown in the following formula. Subsequently, the new global model is distributed to clients again for further local training. This process repeats iteratively until the global model reaches a converged state.

$$w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k \tag{1}$$

Before participating client nodes send their weights to the central server, they can complete different rounds of local model iterations based on their own conditions. It is not necessary for clients to maintain consistency among themselves. Under the assumption of no data heterogeneity, FL algorithms have been proven to exhibit excellent performance and effectiveness.

2.2. The Impact of Client Data Heterogeneity on Model Performance

Due to the inherent data distribution heterogeneity among client devices in practical FL applications, it often leads to instability in training the FedAvg model, resulting in suboptimal performance or even model divergence, making it unable to converge effectively [15]. This happens because when each participating client updates its local model, the model tends to iterate towards optimizing its local objective rather than the global objective. As training iterations progress, the optimization directions between each client's local model and the global objective gradually become inconsistent. This means that local models start to deviate from the global objective. This inconsistency accumulates over time, requiring more communication rounds to bring the global model to a convergent state, further reducing the performance of federated learning as it demands more communication and iterations to accomplish the task. The model iteration trends for local model w_{lk} and global model w_G in two different data distribution scenarios are illustrated in the figure below. In the non-data-heterogeneous scenario, the optimization directions among local models are generally consistent, resulting in no significant deviation from the global model. However, in the data-heterogeneous scenario, due to differences in the local data distribution across clients and the increasing iteration count, substantial deviations in optimization directions occur, thereby affecting the convergence of the global model.

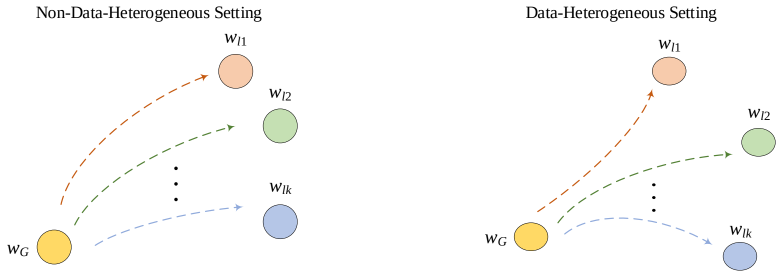

Figure 2. Illustration of FL Client Model Iteration Trends in Different Scenarios

3. METHOD

In practical applications, Federated Learning (FL) involves multiple participant local clients and a central server. These clients may have varying quantities of private datasets with different data distributions, leading to data heterogeneity. FL aims to learn a global model that performs well on all participant client nodes. However, in scenarios with non-IID (Non-Independently and Identically Distributed) and data heterogeneity, aggregation methods like FedAvg can introduce bias into the global model and exhibit limited robustness, ultimately leading to a decrease in FL model performance. Therefore, this paper presents a Federated Learning aggregation algorithm based on data feature information. On the server side, the algorithm first clusters participant client nodes based on their data feature information, assigning different model aggregation weights to clusters of clients with distinct characteristics. On the client side, during the local model iteration process, a regularization term is added to the loss function. This regularization term aims to align the convergence direction of the local model with the global model, preventing the global model from introducing bias. Through experiments, this paper demonstrates the superiority of this algorithm compared to FedAvg and its enhanced robustness.

3.1. The Clustering Method Based on Data Feature Information

In the FL process, participating client nodes need to provide not only their local model parameters to the server but also information about their local data, including the data volume and data distribution. This combined information is referred to as "data feature information." Since this information doesn't involve the actual data itself, it mitigates the risk of privacy breaches for the participating clients. The central server uses the data feature information provided by the client nodes to perform clustering. This clustering process can reduce computational overhead and accelerate the convergence of the global model. The data volume, denoted as q_i, represents the total amount of data for any individual participating client. The data distribution, characterized by the variance Var_i, is determined by the fluctuation range of the local data for that client. The value of Var_i is computed based on the data volumes of various classes within the local client data. The clustering method based on data feature information is outlined in Algorithm 1. Before formally aggregating the global model, the server first conducts a clustering analysis of the data features uploaded by the participating clients. It starts by determining the number of clusters and randomly defining the cluster centroids for each cluster. Then, it proceeds with the clustering calculation. Based on the distance between the client data features and the cluster centroids, the data features are assigned to different clusters. Subsequently, the new cluster centroids are computed as the average of the data features within each cluster. After each round of clustering computation, the centers of each cluster are updated. If the centroids no longer change, it indicates that the clustering is complete, and no further calculations are performed.

Finally, the server will obtain several clusters that represent the participating client nodes based on their local data features. Using clustering to partition the participating client nodes allows for quantifying their contributions to the global model in federated learning. This approach ensures that the global model can be aggregated in a more rational manner, taking into account the characteristics and contributions of each local client.

3.2. The Model Aggregation Method Based on Data Feature Information

Under the condition of data heterogeneity among participants, the FedAvg algorithm does not perform ideally. The FL model aggregation algorithm proposed in this paper is an adaptive weighted aggregation algorithm. Therefore, the most critical issue is how to allocate appropriate weights to participating client clients. As shown in the figure 3, the local data feature information of the participants consists of data quantity and data distribution. In this paper, participants with a large amount of data and a good data distribution are defined as High-Quality Clients, while participants with a small amount of data and a poor data distribution are defined as Low-Quality Clients, and the rest are considered as Standard-Quality Clients, each assigned corresponding weight coefficients $\alpha,\ \gamma,\ \beta \in (0,1)$,where $\alpha \geq \beta \geq \gamma$. According to the different data feature information of local clients, the FL server will use different weight coefficients for global model aggregation.

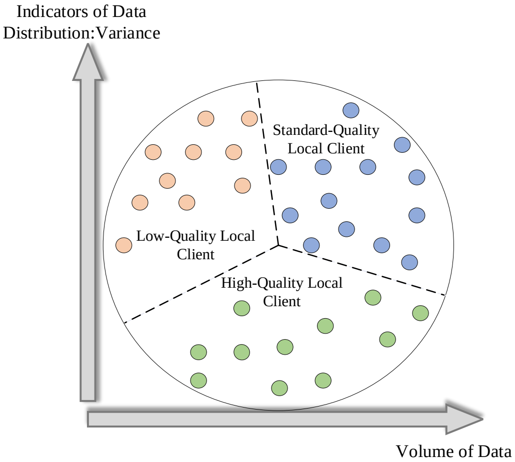

Figure 3. After clustering the FL clients, as shown in the diagram, their distribution is displayed, this paper divides them into three different quality client clusters

3.3. Client Local Model Loss Function Suppression Term

In order to control the alignment of local models and the global model during the FL learning process and avoid potential model divergence, this paper introduces a method that incorporates a suppression term into the client's local model loss function. The loss function plays a crucial role in deep learning, serving as the objective function for optimization in each deep learning model. It guides the model in the direction of better learning through iterations. The FedProx algorithm has shown that adding a regularization penalty term to the local model loss function can reduce the impact of data heterogeneity to some extent. Moreover, the model can achieve convergence performance similar to FedAvg, producing satisfactory results. The target loss function h_k that needs to be optimized is as follows, and the inclusion of the regularization term effectively limits the influence of local model updates on the FL global model.

$$\min_{w} h_k\left(w; w^t\right) = F_k\left(w\right) + \frac{\mu}{2}\left\|w - w^t\right\|^2 \tag{2}$$

Building upon this foundation, this paper takes into account the data feature information of local clients. Clients participating in FL are divided into different levels of client clusters. Therefore, unlike the FedProx algorithm, which adds backend terms from local to global, this paper designs backend terms separately for local to cluster and local to global, thereby restricting the local model loss function. It introduces hyperparameters to control the strength of these penalty terms. The local model optimization objective function l_k proposed in this paper is as follows:

$$\min_{w} l_k\left(w_l; w_c; w^t\right) = F_k\left(w\right) + \frac{\mu_1}{C}\left\|w_l - w_c\right\|^2 + \frac{\mu_2}{N}\left\|w_l - w^t\right\|^2 \tag{3}$$

When FL clients are divided into N clusters, $C > 0$,where represents the number of clients in a particular individual cluster, each client obtains local model parameters w_l after completing local model training. On the server side, the local weights of clients within the same cluster are averaged to obtain the cluster model parameters w_c . Then, weighted aggregation is performed based on the weight coefficients of different clusters to obtain w^t . Additionally, μ_1 and μ_2 are hyperparameters used to control the strength of this term. Due to the integration of clustering, local loss function suppression terms, and more, compared to FedProx, it considers more factors, including the relationships between local and global, cluster and global aspects. Therefore, l_k can better ensure that the local model converges in the same direction as the FL global model, preventing model divergence. This approach performs better in scenarios with heterogeneous client data.

4. EXPERIMENTS AND ANALYSIS

This section describes the experiments conducted to evaluate the proposed method on a dataset of pulmonary TB CT images. Simulations of real-world communication costs are also performed. In order to validate the effectiveness of the proposed method, experiments are conducted in two scenarios: one with heterogeneous client data and one without data heterogeneity.

4.1. Experimental Data Configuration

The pulmonary TB CT image dataset was provided by the People's Hospital of Yuechi County, Sichuan Province, which is a designated hospital for the treatment of TB in the local area. The dataset includes CT images of diagnosed pulmonary TB patients and normal lung CT images from various regions between January 2019 and March 2022. These CT images were categorized by radiologists to distinguish CT images with pathological lesions (indicating TB) from normal CT images without such lesions. Additionally, the data collection process adhered to the following criteria: (1)Patients were diagnosed with pulmonary TB and did not have other pulmonary diseases to prevent interference from other lung conditions in the TB images. (2)The CT images were relatively clear, excluding blurry images with no diagnostic value. (3)The dataset of normal CT images primarily consisted of follow-up CT images taken after the diagnosis of pulmonary TB and subsequent treatment and recovery.

The CT image data was preprocessed to remove personal patient information and to isolate the lung area using image segmentation techniques to eliminate the effects of image noise. The final image data used for the experiments, as shown in Figure 4, only includes the lung region.

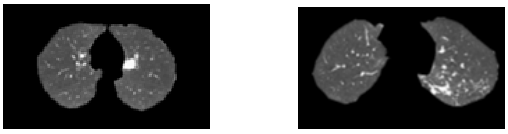

Figure 4. CT images after processing. On the left side are normal lung CT images, and on the right side are CT images of patients with pulmonary TB

To simulate data heterogeneity as closely as possible to real-world scenarios, we assume 10 participating hospitals as FL clients. Basic parameters such as data volume, data categories, etc., are set based on random numbers, and settings related to the Dirichlet distribution are used. The Dirichlet distribution is commonly employed in machine learning to construct Dirichlet mixture models. Its probability distribution function is as shown in Equation 4. By adjusting the value of parameter α , you can control the degree of deviation in the data distribution among FL clients. When α 's value is closer to 0, it indicates greater differences in the data distribution among FL clients, and conversely, a larger α value implies smaller differences.

$$Dir\left(\theta \mid \alpha\right)=\frac{\Gamma\left(\sum_{k=1}^{K}\alpha_k\right)}{\Gamma\left(\alpha_1\right)\Gamma\left(\alpha_2\right)\cdots\Gamma\left(\alpha_K\right)}\cdot\prod_{k=1}^{K}\theta_k^{\alpha_k-1} \tag{4}$$

Furthermore, To introduce data heterogeneity, a random number-based simulation approach was applied. Initially, two sets of random numbers were generated, representing the number of image data samples each client possesses. Then, random numbers were used to select a subset of FL clients, causing them to have only one category of image data, while the remaining clients retained two categories of image data. This method was employed to create a state of data heterogeneity among FL clients. To better mimic real-world scenarios, data shuffling was also conducted. Based on the results obtained from the previous method, FL clients were paired sequentially, with one FL client (denoted as H) exchanging data with any other client (denoted as I) in a random order. The purpose of this data swapping was to increase the degree of label shifting for some FL clients. In the end, we obtained data from 10 FL clients, along with information about data heterogeneity.

4.2. Experimental Model Architecture

The experiment used a convolutional neural network (CNN) model with a total of 4 layers, and ReLU was used as the activation function between the convolutional layers. The overall model structure is shown in the table 1 below.

Table 1. Model Architecture

Structural Name	Type	Size	Output Channels
Conv1	Convolutional Layer	3*3	32

Max_pool	Max Pooling Layer	2*2	32
Conv2	Convolutional Layer	5*5	64
Max_pool	Max Pooling Layer	2*2	64
Conv3	Convolutional Layer	5*5	256
Max_pool	Max Pooling Layer	2*2	256
Fc1	Dense Layer	—	128
Fc2	Dense Layer	—	2

The hyperparameters were set as follows: the global iteration count for FL was set to 100 rounds; Under the assumption of not considering communication costs, the number of participating client clients was set to 10; the local training rounds for each participating client are set to 10; the learning rate η was set to 0.01; batch-size was set to 64; The regularization coefficients μ_1 and μ_2 are set to 0.01 and 0.1, these values are chosen to apply a relatively strong suppression on the differences between the local models and the global model.

4.3. Experiment Results Analysis

4.3.1. The Impact of Data Heterogeneity on FL Model Performance

In this study, the impact of data heterogeneity among FL clients on the global model was examined and elucidated. A convolutional neural network model was trained using a TB CT image dataset and the FedAvg method, as depicted in the following Figure 5. Two scenarios were set up: data heterogeneity and non-data heterogeneity. To ensure fairness, all other training parameters were kept consistent, and scenarios simulating communication costs were not considered. Under the non-data heterogeneity scenario, the FedAvg algorithm exhibited relatively favorable results, achieving rapid model convergence. However, under the influence of data heterogeneity, it was observed that the FedAvg algorithm experienced a significant decrease in model accuracy and convergence speed as the global model continued to train iteratively. Specifically, in the initial stages and the last round of FL, the test set accuracy decreased by 26.41% and 13.05%, respectively.

These experimental findings demonstrate that data heterogeneity among client devices had a substantial impact on FL model performance, resulting in reduced model accuracy and slower convergence speed. Hence, practical FL implementations must take into account the presence of client data heterogeneity.

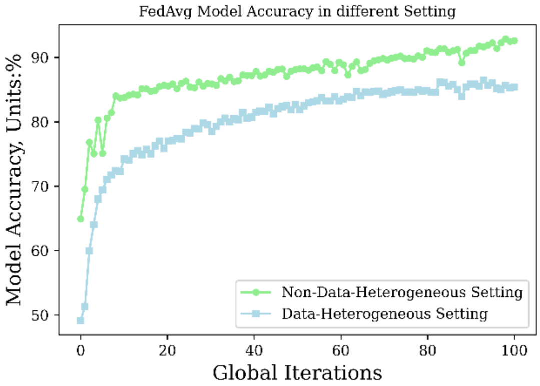

Figure 5. The model performances of FedAvg in different scenarios

4.3.2. Comparison of FL Methods' Performance in Data-Heterogeneous Scenarios

Figure 6 illustrates the trend of model accuracy for FedAvg, FedProx, and our proposed method on the TB CT image dataset in data-heterogeneous scenarios. To eliminate training variability, each method was trained five times, and the average accuracy was computed. Additionally, different random seeds were used for model initialization in each run. Simulated communication costs were introduced, with half of the clients participating in FL selected randomly in each round. From the graph, it is evident that our proposed method outperforms other FL methods significantly. When the global model reaches 100 iterations, our method shows a notable increase in accuracy compared to FedAvg and FedProx. Furthermore, under the influence of communication costs, the model accuracy of FedAvg and FedProx fluctuates significantly in different rounds, demonstrating poor robustness. In contrast, our proposed method exhibits greater stability, particularly after 40 iterations, where the model accuracy remains relatively consistent, indicating a gradual convergence, while the accuracy of FedAvg and FedProx methods continues to vary noticeably.

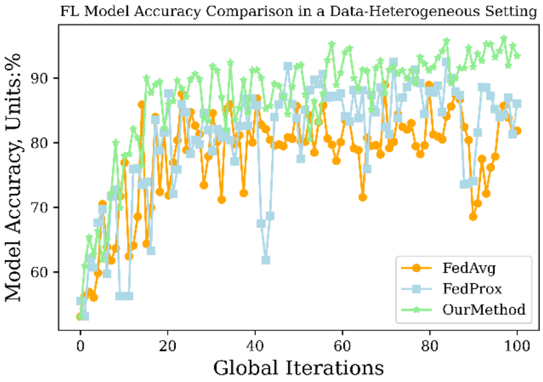

Figure 6. Comparison of Model Accuracy Iteration Trends for FL Algorithms in Data-Heterogeneous Scenarios

4.3.3. Comparison of FL in Non-Data-Heterogeneous Scenarios

To validate the robustness of the proposed method, comparative experiments were conducted in a non-data-heterogeneous scenario, taking into account randomness and communication costs. Since FedAvg algorithm already performs best in non-data-heterogeneous scenarios among FL clients, it was used as the baseline model for comparison. The Figure 7 shows the comparison of accuracy between our method and FedAvg after 100 global model iterations. It can be observed that, overall, there is little difference between the two, and their performance is relatively close in the non-data-heterogeneous scenario. As the FL rounds increase, the models from both methods gradually converge, with a final-round model accuracy difference of only 0.82%.

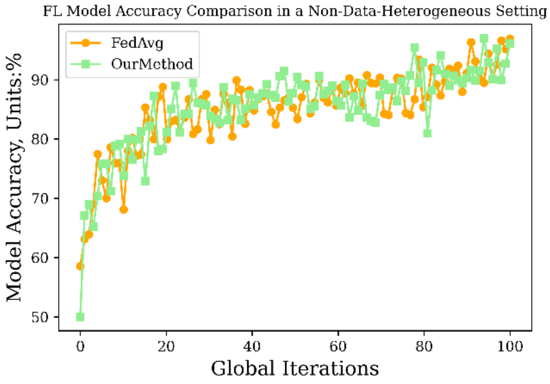

Figure 7. In a non-data-heterogeneous scenario, the comparison between the method proposed in this paper and FedAvg

4.3.4. Presentation of Experimental Results

The table 2 shows the comparison of our method with other FL baseline methods in terms of accuracy on the pulmonary TB CT image test set in both data-heterogeneous and non-data-heterogeneous scenarios. From the table, it can be observed that in the non-data-heterogeneous scenario, our method performs similarly to the FedAvg algorithm. In the data-heterogeneous scenario, our method outperforms other FL baseline methods. When the communication rounds reach 100, the accuracy is improved by 11.54% and 7.38% compared to FedAvg and FedProx, respectively. Therefore, the experimental results demonstrate good model performance in both data-heterogeneous and non-data-heterogeneous application scenarios.

Table 2. Comparison of Model Accuracy Presentation

Dataset	Data Heterogeneity Settings	Communication Rounds	Test Accuracy/%		
			FedAvg	FedProx	OurMethod
TB CT image dataset	Non-Data heterogeneity scenario	40	88.26	—	87.57
		60	85.58	—	86.43
		80	85.39	—	92.92
		100	96.91	—	96.08
	Data heterogeneity scenario	40	80.05	86.89	91.21
		60	80.10	87.57	90.81
		80	88.91	87.99	91.15
		100	81.86	86.02	93.40

5. CONCLUSION

In real-world applications, FL often encounters the problem of data heterogeneity among participating client devices, and this is also the case in the medical field. To address this issue, this paper proposes an FL approach that differs from the averaging aggregation method used by the FedAvg algorithm. Instead, it employs weighted model aggregation based on client clustering. This approach enhances the contribution of high-quality clients to the FL process while reducing interference from low-quality clients. Additionally, it designs a local model loss function suppression term to control the potential model deviation during the local and global model iteration processes, ensuring that their convergence directions remain as consistent as possible. Experimental results demonstrate that this method performs well in the presence of data heterogeneity, achieving high classification accuracy on the used dataset of pulmonary TB CT images.

REFERENCES

[1] Rahman M T, Codlin A J, Rahman M M, et al. An evaluation of automated chest radiography reading software for TB screening among public-and private-sector patients[J]. European Respiratory Journal, 2017, 49(5).

[2] Pande T, Cohen C, Pai M, et al. Computer-aided detection of pulmonary TB on digital chest radiographs: a systematic review[J]. The International Journal of TB and Lung Disease, 2016, 20(9): 1226-1230.

[3] Wang Xuesong, Dong Hanbin, Zhong Cheng. Relationship between sputum culture results and CT features of secondary active pulmonary tuberculosis in adults [J]. Chinese Journal of Integrative Imaging,2023,21(04):424-427.

[4] Pan Jishu. Pay attention to the imaging diagnosis of pulmonary tuberculosis [J]. Chinese Journal of Radiology,2004(01):5-6.

[5] Yang A, Jin X, Li L. CT images recognition of pulmonary TB based on improved faster RCNN and U-Net[C]//2019 10th international Conference on information Technology in Medicine and education (ITME). IEEE, 2019: 93-97.

[6] Chen Chao-yi, Xu Bo, Wu Ying et al. A review of attention mechanism in medical image processing [J]. Computer Engineering and Applications,2022,58(05):23-33.

[7] MARTINI K, LOUBET A, BANKIER A, et al. Nodular reverse halo sign in active pulmonary TB: a rare CT feature? [J]. Diagnostic and Interventional Imaging, 2020, 101(5): 281 – 287

[8] Hu K, Li Y, Zhang S, et al. FedMMD: A Federated weighting algorithm considering Non-IID and Local Model Deviation[J]. Expert Systems with Applications, 2023: 121463.

[9] Gavrilov A D, Jordache A, Vasdani M, et al. Preventing model overfitting and underfitting in convolutional neural networks[J]. International Journal of Software Science and Computational Intelligence (IJSSCI), 2018, 10(4): 19-28.

[10] Yang Q, Liu Y, Chen T, et al. Federated machine learning: Concept and applications[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(2): 1-19.

[11] Ye Ming, Wang Yan. Research on the legal system of Data islanding in the Era of Artificial Intelligence [J]. Journal of Dalian University of Technology (Social Science Edition),2019,40(05): 69-77. DOI:10.19525/j.issn1008-407x.2019.05.009.

[12] Li Q, Diao Y, Chen Q, et al. Federated learning on non-iid data silos: An experimental study[C]//2022 IEEE 38th International Conference on Data Engineering (ICDE). IEEE, 2022: 965-978.

[13] Zou Chengming, Zhao Ning. Correction and aggregation of federated learning model in heterogeneous data scenarios [J]. Electronic Measurement technology,2022,45(20): 102-109.DOI:10.19651/j.cnki.emt.2209707.

[14] Li X, Huang K, Yang W, et al. On the convergence of fedavg on non-iid data[J]. arXiv preprint arXiv:1907.02189, 2019.

[15] Zhao Y, Li M, Lai L, et al. Federated learning with non-iid data[J]. arXiv preprint arXiv:1806.00582, 2018.